34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

Quantifying Hardware Needs Prior to Production

Spares Strategy for Programs in Development Phases

Outline

- Problem Context
- Spares Strategies
- Proposed Approach
- Case Study
- Observations
- Future Work

Problem Context

Identification of Spares

- Sparing strategies
  - Ensure spare parts are available for unplanned failures
  - But no surplus of unneeded parts
- Systems in **production or operation**
- Procured in advance
  - Available for use in the event of a failure
- Overhead of procurement and storage offsets the risk of down time if no spare is available

How are Development Programs Different?

- Design in progress
  - Failures
  - Changes in design of parts

- Program delays due to lack of parts can be significant
  - Too few or wrong parts can delay activities crucial to finalizing design
  - However
    - Difficult to know which parts will be needed
    - Spares may become outdated or never be used
    - Use of parts that are of incorrect pedigree (design version) can compromise results

Challenges in Defining Spares for Development Programs

- What parts are needed?
  - How to determine how many parts of each version of the design and when?
  - Ensure ability to continue activities when expected failures occur
  - Reduce waste and inventory management cost
- Limited data to make decisions
  - Lack of guidance to make informed assumptions and decisions
  - Many programs choose spares based on experience of individuals working program
    - Varying degrees of success
  - Difficult to justify budget and/or schedule requests

Current Approaches

Spares Strategies

Production Sparing Strategy

Well Defined

- Tied to reliability analyses
- Considers:
  - Delivery time
  - Cost
  - Downtime
  - Lost production costs
  - Failure and repair data
- Calculates the quantity of spare parts needed at any specific time

Calculated as Cost VS Risk

The goal: reduce risk to an acceptable level with an acceptable cost.

$$\text{Cost}=\sum_{i=1} S_i + O_{Si} \quad (1)$$

$$\text{Risk}=\sum_{i=1} P_{fi} * C_{fi} \quad (2)$$

- S, Cost of the spare(s)
- O_S, Associated overhead for the spares (inventory, maintenance and personnel costs), for each spare (i).
- P_f is the probability of failure
- C_f the consequences of the unwanted event

Proposed Approach

Development Sparing Strategy

Proposed Definition

- Calculation of cost and risk is more difficult
- Cost
  - Cost of part – may include Non- recurring engineering
  - No long-term inventory or overhead
  - Potential cost to modify spare
  - Spares may not be used
- Risk
  - Probability of failure is based on design maturity
  - Consequence of a failure is related to impact on design progress if a spare is not available

Calculated as Cost VS Risk

Cost=$\sum_{i=1} = S_i + M_{fi}$ (3)

Risk=$\sum_{i=1} P_{fci} * C_{fdi}$ (4)

- *S*, Cost of the procurement of the spare
- M_f, Potential cost to modify the spare,
- P_{fd}, is the probability of failure in development system
- C_{fd}, Consequence of failure in development system

Calculation of the Cost of Consequence

$$C_{fd} = (D+D_p)*E*R*N*I$$

- *D* - duration of the delay (to replace the part if no spare), Program daily execution cost * number of days delay + any retest costs (facilities, etc)
- D_p,- the potential damage to system (dollars for repair/rework – estimate)
- *E* - a factor representing the criticality or essential nature of the part (can tests continue without the part?). 0 for a part that is rarely used. 1 for all other modules/cables except for security essential and 2 for security essential modules/cables.
- *R* - factor representing the redundancy in the system (are there other parts that can be used while a replacement is found?). 0 for more than 3 modules/cables redundancy, 1 for 2 modules/cables redundancy, 2 for 1 or less modules/cables redundancy.
- *N* - the state of the design of the part reflecting the necessity (are there factors indicating the part cannot or will not change and a representative can replace the part such as a load or an emulator?). 0 for not necessary or a replacement emulator or load is sufficient and 1 for necessary.
- *I* - factor representing the resulting impact on major program milestones as the result of a failure (specifically the First Production Unit (FPU) or the Initial Operating Capability (IOC)). 1 for no push of FPU or IOC, 2 for push of FPU or IOC.

Case Study

Case Study

- Well Defined Program Plan/Schedule
  - Two prototype builds before final design for production
  - Environmental test can be at system or subsystem level
  - System and subsystem functional test beds
  - Testers available for subsystem and system test
  - Full system level test with each prototype build
  - Normal environments and destructive abnormal environments
  - Schedule defined - No spares allocated

Case Study

- Well Defined Program Plan/Schedule
  - Tw roduction
  - Env stem level
  - Sys
  - Tes est
  - Full system level test with each prototype build
  - Normal environments and destructive abnormal environments
  - Schedule defined - No spares allocated

Defined budget
Defined schedule
Defined # of parts

Case Study

- Well Defined Program Plan/Schedule
  - Tw… roduction
  - En… ystem level
  - Sy…
  - Tes… est
  - Ful… d
  - No… nal env…
  - Schedule defined - No spares allocated

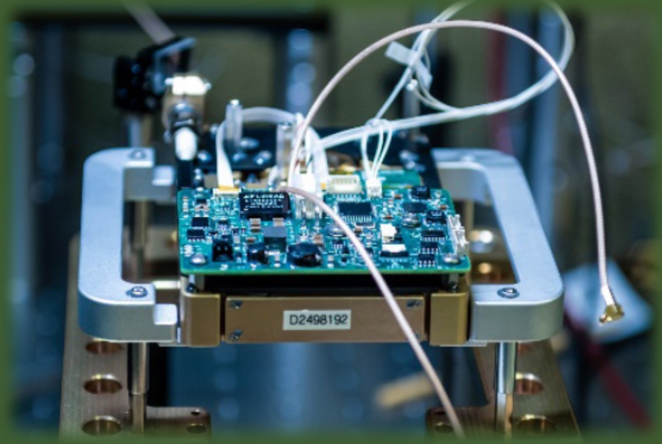
Defined budget
Defined schedule
Defined # of parts

?? Spares ??

Case Study

- Calculation of Cost VS Risk needed to be narrowed down
- Program had defined risks and risk contingency funds
  - Some risks related to failures in development schedule
  - Delayed or repeated tests
  - Re-spin on designs
  - Alternate designs
  - Focus on electronic design and cables; mechanical structure design mature

Case Study

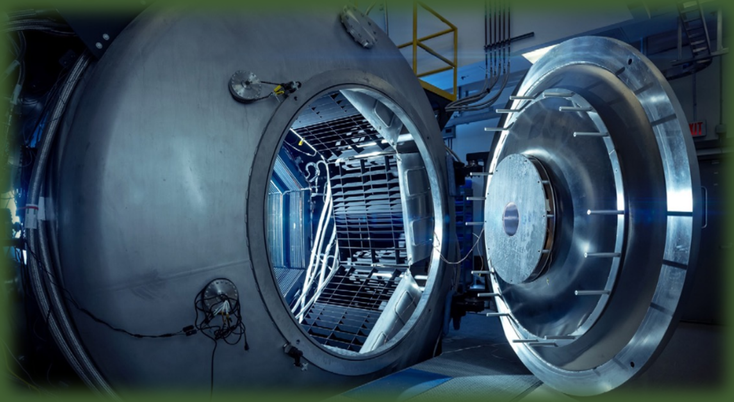


- General Rules to narrow problem space
  - Readily available COTs = no spares
  - Mass mocks for subsystems available
  - Quantities for environmental qualification = 3
  - Cables – connectors/wires can be spares but don’t build cables
  - PWBs – parts versus populated boards
  - One of a kind items priority for spares
  - Critical parts priority for spares
  - Assume 20% failure rate**
  - Consequence can affect schedule, damage to system if part fails, cost, technical integrity of data, etc. – converted to dollars

Case Study

- Development activities:
  - Subsystem functional tests, Prototype 1 and Prototype 2
  - Subsystem environmental tests, Prototype 1 and Prototype 2
  - System functional tests, Prototype 1 and Prototype 2
  - System destructive environmental tests, Prototype 1 and Prototype 2

Case Study



Example: System Level Destructive Test

- Cost: 40 subsystems at $55k each + $10k modification cost for each = $2.6M
- Risk:
  - Parts may fail and need to be replaced – with no spares, there is a 6 month delay to procure, checkout and install spare. Test results required to move to next phase.
  - 20% probability of a single subsystem failing
  - 6 months (at $50M yr burn rate = $25M + cost of spare $55k) = $25.06M
  - Risk = $5M

Cost < Risk

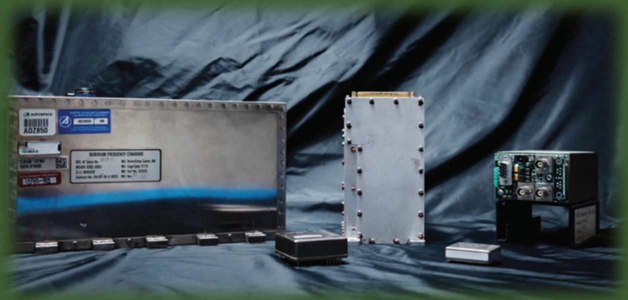
- Residual opportunity – spares not used would be available for other activities

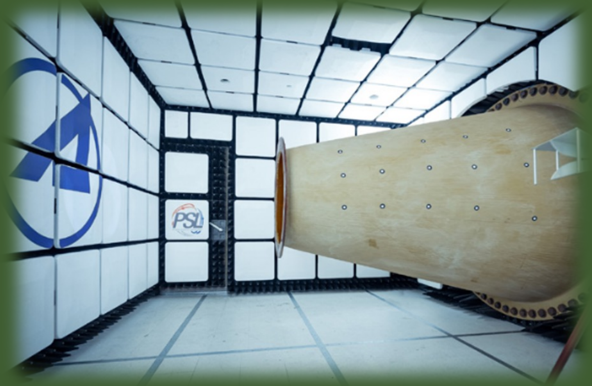
Calculation of the Cost of Consequence

$$C_{fd} = (D+D_p)*E*R*N*I$$

- *D* - duration of the delay. Program daily execution cost * number of days delay + any retest costs (facilities, etc)
- D_p,- the potential damage to system (dollars for repair/rework – estimate)
- *E* - a factor representing the criticality of the part (can tests continue without the part?). 0 for a part that is rarely used. 1 for all other parts except for critical and 2 for critical parts.
- *R – Redundancy* factor (are there other parts that can be used while a replacement is found?). 0 for more than 3 part redundancy, 1 for 2 part redundancy, 2 for 1 or less part redundancy.
- *N* - Necessity of part (a representative can replace the part such as a load or an emulator). 0 for not necessary or a replacement emulator or load is sufficient and 1 for necessary.
- *I – Impact* factor - impact on major program milestones as the result of a failure (specifically the First Production Unit (FPU) or the Initial Operating Capability (IOC)). 1 for no push of FPU or IOC, 2 for push of FPU or IOC.

Case Study





Example: Subsystem Environmental Tests

- Cost: 1 subsystem at $55k each + $10k modification cost for each = $65K
- Risk:
  - Parts may fail and need to be replaced – there are 3 units, one may be repurposed from a different environment while a spare is obtained (6 months or part is repaired/modified 1 month).
  - 20% probability of a single subsystem failing
  - Consequence is delay of information to next design spin – margin in schedule. No schedule impact, added expense of repair or spare ($55k +10k mod) = $65k
  - Risk = $13K

Risk < Cost

- Residual Risk – prototype unit may be exposed to more environmental testing than planned – consider impact to later testing

Observations

Process Flow

Development Program

Sparing Strategy.

01	Define Program Constraints	Schedule, Budget, activities, program costs, assumptions
02	Define Assumptions	Critical or Priority Items, Items not needing spares, failure probabilities based on maturity
03	Assess each activity	Perform Cost VS Risk assessment for each activity. Phased approach
04	Assess Residual Risk / Opportunities	Consolidate spares if possible; identify risk reduction activities
05	Prepare a spares schedule	Document plan – including plan for unused spares. Identify when spares need to be procured based on activity need.
06	Reassess spares strategy	Any program changes could drive a spares strategy update
07	Lessons Learned	Keep historical data.

Observations

- Effectiveness of Process
  - Must work within constraints of program (HW plans, schedule, design needs)
  - Program plan changes = spares strategy must be revisited
  - Most effective when assessing each planned activity separately, then revisit total number of spares for sharing opportunities
  - Provided basis for allocating risk funds or management reserve
  - Drove decisions on order of activities
  - Could not account for every possible failure
- Ease of Process
  - Required great familiarity with HW, Schedule, and design assessment activities
  - Required Program cost data or estimates
  - Required understanding of typical types of failures
  - Would require a chief engineer or lead Systems Engineer to execute

Future Work

Future Work

- Additional Areas of investigation
  - Capturing residual risk/opportunity
  - Prédictive failure rates based on common technology
  - Criteria for ideal ratio of Risk to Cost
  - Leveraging technology readiness assessments (TRL/MRL) in calculations
  - Decision tree/process flow chart/automated application
- Industry case studies
  - Transportation
  - Aircraft
  - Industrial equipment
  - Medical
  - Satellites

Questions?

Contact Info

Davinia Rizzo, Ph.D.
Aerospace Corporation
davinia.rizzo@aero.org

34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

www.incose.org/symp2024
#INCOSEIS